(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 December 2002 (05.12.2002)

PCT

(10) International Publication Number
WO 02/097584 A2

(51) International Patent Classification[7]: G06F

(21) International Application Number: PCT/US02/17118

(22) International Filing Date: 31 May 2002 (31.05.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/294,835	31 May 2001 (31.05.2001)	US
Not furnished	30 May 2002 (30.05.2002)	US

(71) Applicant: **HYPERSPACE COMMUNICATIONS, INC.** [US/US]; 8375 South Willow Street, Suite 400, Littleton, CO 80124 (US).

(72) Inventors: **CRANSTONE, Peter J.**; 10721 Monaco Street, Littleton, CO 80124 (US). **KILEY, Kevin J.**; 473 Baylor House Road, Bigelow, AR 72016 (US).

(74) Agents: **LAWRENZ, Steven D.** et al.; Perkins Coie LLP, P.O. Box 1247, Seattle, WA 98111-1247 (US).

(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZM, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

WO 02/097584 A2

(54) Title: ADAPTIVE VIDEO SERVER

(57) Abstract: A client-server system includes a client configured to display video wherein the client is coupled to a network, and a server that is also coupled to the network. The server is configured to determine computing resource characteristics of the client. The server is also configured to select a transmit video format from a plurality of available video formats based on the determined computing resource characteristics of the client. The server is then adapted to communicate video data to the client device in the selected transmit video format.

ADAPTIVE VIDEO SERVER

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims the benefit of U.S. Provisional Patent Application "ADAPTIVE VIDEO SERVER," No. 60/294,835, filed May 31, 2001 (attorney docket number 36120-8001US00).

BACKGROUND

[0002] Various embodiments relate generally to a multimedia communications system, and, more particularly, to a communications system having an adaptive video server.

[0003] There has been much investigation into improvements in the distribution of multimedia content over networks for varied purposes such as entertainment, education and the like. While initially multimedia was distributed mainly over terrestrial broadcasting channels (*e.g.,* network television), the distribution channels have evolved to include community access television (CATV, a.k.a. "cable" TV), digital satellite distribution, and the Internet.

[0004] Sophisticated digital transport networks have been developed, generally, for each of the above channels. Particularly, protocols have been developed for the transport of multimedia content over the Internet. For example, documents entitled *Experimental Internet Stream Protocol, Version 2*, RFC 1190 (October 1990; available at http://www.rfc-editor.org/go.html) and *Real Time Streaming Protocol (RTSP)*, RFC 2326 (April 1998; available at http://www.rfc-editor.org/go.html) describe such standards, both herein incorporated by reference. The *Real Time Streaming Protocol (RTSP)* document discloses an application-level protocol for the control over the delivery of data with real-time properties, such data including audio and video sourced from either live feeds and stored clips. The *Real Time Streaming Protocol (RTSP)* document further contemplates choice from among a variety of Internet delivery channels (such as UDP, multicast UDP and TCP). These

protocols, however, only govern *how* the data is transferred, not *what* the data should be (*i.e.,* in what format the multimedia should be packaged).

[0005] One prevailing multimedia format, namely the Motion Picture Experts Group (MPEG) format, is disclosed in exemplary fashion in U.S. Patent No. 6,148,142 issued to Anderson entitled "MULTI-USER, ON-DEMAND VIDEO SERVER SYSTEM INCLUDING INDEPENDENT, CONCURRENTLY OPERATING REMOTE DATA RETRIEVAL CONTROLLER". Anderson discloses a video server having a mass storage unit for storing a plurality of movies in digital form, specifically in an MPEG format. Anderson further discloses a plurality of identical converters, one for each conventional television receiver, for decoding and decompressing the MPEG movies. The link between the video server and the converters is disclosed as being high bandwidth. There are several disadvantages with the video server of Anderson.

[0006] One limitation is that is requires *identical* MPEG converter devices. This requirement limits the utility of the video server to use only over closed networks where end-user devices can be controlled to be the same. Another limitation is that is requires a high bandwidth connection between the video server and the converter. With the large number of dial-up analog modem users (*e.g.,* 56 kbps), and particularly in view of the gaining acceptance of wireless connections (*e.g.,* 9.6 kbps), this requirement also limits the potential audience the video server may service. Yet another disadvantage is that is requires sufficient computing power in the converter to decode and decompress MPEG video in real time. Many handheld computing devices scarcely have the memory to store an MPEG codec, much less the computing power to decode and decompress in real time. In addition, even were the user willing to accept substantially less than real time playback (*e.g.,* a frame every 20 seconds), such use would quickly run down the handheld's battery. Again, the video server of Anderson has utility only for a limited audience of devices.

[0007] One approach taken in the art ostensibly to address the problems that arise when delivering multimedia to low memory/computing resource devices (*e.g.,* Palm connected organizers, Pocket PCs, and the like) involves packaging the content in a proprietary format for playback using a proprietary viewer so as to allegedly improve on the frames/per/second playback speed. Exemplary of this approach includes a product available under the name FIREPRODUCER (http://www.firepad.com), which purports to deliver streams of live or stored video to multiple PalmOS handheld devices that run a client piece called FIREVIEWER. However, the output of this product is inflexible inasmuch as the proprietary video format is generally incompatible with other devices, or, at the very least, is not optimized or tailored to suit end user devices other than PalmOS.

[0008] Another approach, for example Microsoft's *Netmeeting* product, requires several "hard" connections be opened (*i.e.,* ports) on the destination client device, which also must run a Microsoft Windows-based operating system. This makes such a product difficult to use through a firewall or where hard connections as described above are not available or permitted.

[0009] Another approach taken in the art to address the problem of variable end-user devices is seen in U.S. Patent 5,905,524 to Sauer entitled "DIGITAL ISDN VIDEO SERVER". Sauer discloses a video server that provides video and audio signals with different video formats and different transmission rates for different video telephones. Sauer further discloses that the different formats allow the selection of a signal quality which is suitable for a subscriber, and which can operate in accordance with the known H.261 ITU standard. The H.261 ITU standard provides for two resolution choices, with a spectrum of bit rate choices (all of which are multiples of 64kbits, particularly suited for ISDN). The system of Sauer, however, is compression-based and thus presumes significant resources in the end-user device as well as an ISDN connection.

[0010] There remains, therefore, a need for an video server that minimizes, overcomes and/or eliminates one or more of the problems set forth above.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] Figure 1 is a simplified block diagram view of a system according to the invention.

[0012] Figure 2 is a simplified block diagram showing, in greater detail, an architectural organization of the video server of Figure 1.

[0013] Figure 3 is a simplified diagrammatic and block diagram view showing, in greater detail, the portions of the server software illustrated in Figure 2.

[0014] Figure 4 is a simplified block diagram of a client computing architecture according to the invention.

[0015] Figure 5 is a simplified block diagram view of an originating source of video data according to the invention.

[0016] Figure 6 is a simplified diagram of a display buffer of Figure 5.

DETAILED DESCRIPTION

[0017] Various embodiments of a video server adapt video data to a video format that is optimized for play back at a client device based on a number of factors. Another embodiment provides, with respect to an originating source of video data, for image acquisition and communication to a video server using computing devices having limited computing resources.

[0018] A client-server system may include a client configured to display video and which is coupled to a network. The system also includes a server coupled to the network, and which is configured to determine computing resource characteristics of the client. The server is further configured to select a transmit video format from a plurality of available video formats based on the determined computing resource characteristics of the client, and to communicate the video data to the client in the selected video format.

[0019] In one embodiment, the selected transmit video format corresponds to a native display format associated with the client. This is useful in relieving the client with decoding, and decompression functions where the client has relatively limited central processing unit (CPU) computing ability, amounts of main memory, amounts of available memory, or the lack of a specialized

graphics processor. In an alternative embodiment, the server is also configured to translate, for example, color video data for display on a grey-scale display and/or to scale a horizontal and vertical display size so as to fit on the client display.

[0020] Other objects, features, and advantages will become apparent to one of ordinary skill in the art from the following detailed description and drawings which illustrate the invention by way of example, but not by way of limitation.

[0021] Referring now to the drawings wherein like reference numerals are used to identify identical components in the various views, Figure 1 shows a client-server system 10 for multimedia communications in accordance with one embodiment of the present invention.

[0022] System 10 features an adaptive video server configured to select a transmit video format from a plurality of available video formats based on determined characteristics of the destination client. As described in the Background, a variety of end-user devices are available in commerce, each having unique operating and display characteristics, and for which certain video formats are more desirable and suitable than others.

[0023] Figure 1 further shows a video server 12 which may optionally have a stored multimedia database 14 associated therewith, and a plurality of client devices 16_1, 16_2, 16_3, . . ., 16_N connected to the video server 12 by way of a first network 18. Network 18 may include a wireless communication portion 18_W for communications with wireless clients such as client 16_1. Figure 1 further shows a plurality of originating video sources 20_1, . . . , 20_N connected to video server 12 by way of a second network 22. First network 18 and second network 22 may be the same network. As used herein, video means multimedia content including moving video, still video, audio, or a combination thereof.

[0024] Video server 12 is configured to communicate either (i) pre-stored multimedia content, such as contained in its local database 14, or (ii) streaming multimedia content received from originating sources 20, to client devices 16 that have connected therewith and have requested content. Video

server 12 may be implemented on conventional computing hardware. For purposes of example only, server 12 may comprise a personal computer (PC) platform executing a windows operating system (*e.g.*, Windows 98SE, Windows ME, Windows 2000, etc.), an Apple Macintosh computer system, a Unix-based computer, a Linux-based computer, and the like. As such, video server 12 includes conventional central processor(s), main memory, cache memory, hard disk drive, communication ports, keyboard, mouse, display and the like. In addition to the functionality described herein, video server 12 may further include conventional server software, for example, conventional web server software such as the widely available Apache web server software for Linux-based systems (*e.g.*, recently available Apache 1.3.19 version available at http://httpd.apache.org/). It should be understood, however, that server 12 need not comprise a web (or HTTP) server where, in particular, the client(s) 16_1, 16_2, 16_3, . . . , 16_N are not communicating with server 12 using the HTTP protocol.

[0025] The stored multimedia database 14 may comprise any type of content (*e.g.*, entertainment, educational, etc.) in any one of a number of known video formats including, without limitation, the Motion Picture Experts Group (MPEG) format (*e.g.*, MPEG-1, MPEG-2, MPEG-4, etc.), Microsoft video format-AVI format, Quick Time format, or any other type of format now known, or hereinafter developed. Stored multimedia database 14 may reside, physically, on a hard disk drive, or other conventional apparatus (*e.g.*, CD or DVD tower).

[0026] Client devices 16_1, 16_2, 16_3, . . . , 16_N are each configured to display video and are, as shown, coupled to network 18. Certain of the client devices 16 may include a client module to enhance the exchange of information between client 16 and server 12, the content of which will be described in greater detail in connection with Figure 3. For example, client devices 16_2, 16_3 are shown having such a client module, while client devices 16_1 and 16_N do not have such a client module. However, all clients have, at a minimum, capabilities to display video, as mentioned above. The particular

configuration of the plurality of client devices 16 varies greatly, and may be any one of a plurality of conventional computing devices commercially available. For purposes of example only, client devices 16 may comprise a desktop or laptop personal computer (PC) executing a windows-based operating system (*e.g.*, Windows 98SE, Windows ME, Windows 2000, etc.), an Apple Macintosh computer, a Unix work station, a Linux work station, a hand held computer such as a Microsoft Windows CE-based device (*i.e.*, a Pocket PC), a Palm or Hand Spring connected organizer, any variety of laptop computers, wireless telephones having internet access (*e.g.*, WAP enabled), such as, for example, a commercially available product from Kyocera (Webphone), video tablets, wireless wrist watches (*e.g.*, commercially available from Matsucom), Internet appliances (*e.g.*, 3COM's "Audrey"), as well as a variety of other computing devices, generally. The client devices may be either connected to network 18 by way of a wire line, such as shown for client devices 16_2, 16_3 and 16_N or may be connected to network 18 by way of a wireless link, as for the case of client device 16_1 connected to wireless portion 18_W of network 18 (connection shown in phantom line).

[0027] As should be understood, especially in view of the previous description of the range of devices for client device 16, not all video formats may be practically played back on any particular client device 16. Video server 12 may select one of a plurality of video formats based on a profile of the requesting client device 16. The information contained in the profile will be described in connection with Figure 3.

[0028] Referring now to Figure 2, a simplified block diagram view of video server 12 is shown. As illustrated, video server 12 includes an operating system 24, server software 26, and, optionally, one or more virtual machines 28_1, 28_2, . . ., 28_N which may be established by server software 26 depending on the assessment of the profile of a requesting client device 16. In particular, it is quite often the case that content stored in multimedia database 14 is stored in a compressed form (*e.g.*, MPEG-2) in order to reduce the amount of disk space taken up by the content, for example, multiple movies. However,

many end-user client devices 16 have neither the computing resources to decode and decompress an MPEG-2 stream in real time, nor have the appropriate display to show the decoded and decompressed MPEG-2 video even after such processing has been completed. For example, the original movie may be "full screen" and in color, while the client display may be a much smaller grey scale screen. Accordingly, server software 26, when it determines that a change in video format is desirable for a particular client device 16, establishes a respective virtual machine 28 in the server memory itself, where the requested multimedia content is played back/translated, or otherwise formatted so as to optimize the experience for a user of the requesting client device. A similar construct is known in the art and is referred to as a "thin client" approach. The actual processing power, therefore, is drawn from the server, not the client.

[0029] Figure 3 shows, in greater detail, the server software 26 according to one embodiment of the invention. As shown in block, functional form, server software 26 includes input/output (I/O) 30, a client identity detect module 32, a client computer resource characteristic detect module 34, a client display characteristic detect module 36, a server-to-client bandwidth detect module 38, a client battery detect module 40, a video format select module 42, and a video data formatter module 44. Video data formatted in a transmit video format is output as a stream of messages 45. A respective client profile is developed using the detected client information.

[0030] Client identity detect module 32 is configured to detect an identity of a particular client device 16 and generate a signal S0 indicative of such determined identity. For example, such identification information may include the client device manufacturer, model number, operating system, and any special hardware or software. It should be appreciated that signal S0 is shown simply as a single signal line for clarity. Of course, the identification may, in-fact, comprise, for example, text strings corresponding to the above described information, as well as perhaps numerical information.

[0031] Computing resource characteristic detect module 34 is configured to obtain information useful to assess, generally, the capability of a client device 16 to decode and decompress a multimedia content stream which, as described above, is commonly compressed. As used herein, computing resource characteristics may include the type and clock speed of a central processing unit (CPU), including any other characterizing computing ability (*e.g.*, MMX capability), an amount of main memory, an amount of available memory, as well as an availability of any specialized graphics processor. Module 34 determines one or more of the foregoing characteristics and generates a signal, designated S1, indicative of such information. As with signal S0, it is shown as "single" line for clarity only.

[0032] Module 36 is configured to determine display characteristics of a client device 16. Display characteristics as used herein may include the availability of a color display and a color depth associated with each pixel, an availability of a grey-scale display and the number of levels of grey associated therewith, the availability of a black/white display, and a horizontal and vertical display size (*e.g.*, in pixels). In some instances, determination of the identity of the client uniquely determines its display capabilities (*e.g.*, a Palm IIIx is a 160 x 160 pixel, 4-bit grey scale display). Display characteristic detect module 36 generates an output signal S2 indicative of the determined display characteristics of a particular client device 16. Again, signal S2, along with signals S3, S4 and S5, are shown as a single line for clarity purposes only.

[0033] Bandwidth detect module 38 is configured to detect an available bandwidth between video server 12 and a requesting client device 16 through network 18. Bandwidth detect module 38 generates a signal, designated S3 indicative of the determined available bandwidth. In addition, connectivity characteristics such as delay, turnaround time, whether the communication is "connectionless" or not (*e.g.*, TCP/IP and opened ports, etc. vs. video data tunneled via HTTP), may also be included in the information in signal S3.

[0034] Battery detect module 40 is configured to determine whether a particular client device 16 is powered by way of batteries, and if so, the type

and optionally the state of charge thereof. Battery detect module 40 generates an output signal, designated S4, indicative of the determined battery characteristics of the particular client device 16.

[0035] As described above, in one embodiment, client devices 16 include a client module that is configured to communicate and cooperate with video server software 26 of video server 12 in obtaining the foregoing information. In such an embodiment, as shown in Figure 4, such client software 46 is configured to communicate with, for example, application programming interfaces (APIs) 47 of the operating system of the client device 16 in order to ascertain one or more of the items of information described above with respect to modules 32, 34, 36, 38, and 40. For example, in the case of a Palm OS based handheld computer, a set of publicly known APIs are available to determine a wide variety of client device information, such as the information set forth above. In this regard, calling the publicly documented APIs to obtain the aforementioned information requires no more than routine application of programming capabilities of one of ordinary skill in the art and therefore need not be described in detail. For further illustration purposes only, a Palm software development kit (*e.g.*, SDK 3.5) is freely available and widely distributed, for example, at http://www.palmos.com. Similar software development kits are available for other devices (*e.g.*, Microsoft Windows CE APIs for a pocket PC device, Windows 98 APIs for a desktop or laptop client device, etc.).

[0036] As a matter of general principle, the aforementioned information associated with a particular client device 16 may affect the selection of a video format. A general overview of the selection criteria will now be set forth, with concrete examples to follow.

[0037] As to the identity signal S0, such information may be dispositive in selecting a transmit video format. For example, a particular client device 16 may comprise a Microsoft Windows-based operating system. In such case, a windows video format (*e.g.*, a file having a *.avi extension) may be used inasmuch as it has been determined that the windows operating system video

drivers prefer incoming video data via an *.avi format (*i.e.*, as opposed to an MPEG format or a Quick Time format for example). The client identity may also uniquely determine some computing resource characteristics and/or screen display (as mentioned above).

[0038] The computing resource characteristics (*i.e.*, signal S1) largely reflect the client device's capability to decode and decompress certain synchronized and compressed video formats such as the MPEG-2 format, and Real Time Video (RTV) format. For example, it is contended in the literature that 17 frames per second are needed by the human visual cortex to assimilate real time video. Depending on the client device's display characteristics, and the available bandwidth through the network to the client device, the video server 12 may determine that no compression should be used in order to provide the best experience to the user of that particular client device 16 in terms of frames per second, for example. This means that a bitmap video format would be sent.

[0039] The display characteristics (signal S2) may also be considered together with the other signals in determining what transmit video format to select. For example, a movie stored in multimedia storage database 14 in an MPEG-2 format, that is in a color format, in a 1024 x 768 pixel size, for example, will not only have to be decoded and decompressed at the client device prior to display, but may also have to be converted from color to grey-scale, as well as to be scaled down to the display window of the client device (*e.g.*, 160 x 160 pixels in a Palm OS connected organizer). Sufficient computing resources may be unavailable to perform the decoding, decompression <u>*and*</u> *the color-to-grey scale and scaling conversions* in a suitably high frame rate. In such case, the server software may perform the color-to-grey scale and size scaling transformations on the server side and send that in either compressed or uncompressed format to the client device 16.

[0040] The available bandwidth, designated S3, may impact selection of a video format as follows. For example, if a provisional selection by server 12

(perhaps based on other factors) of a video format would be to send uncompressed multimedia content to the client device 16, but the connection is bandwidth limited, for example over a wireless link, then a compromise must be made. The compromise reconciles (i) attempting to optimize frames per second at the client by reducing computationally intensive decoding and decompression (which may unfortunately increase the total payload), and (ii) total bandwidth from server to client (which may be limited and insufficient to carry uncompressed data). That is, some level of compression may be required in order to fit the video data within the maximum available bandwidth. A video format having some level of compression would be selected (*e.g.*, MPEG-4), and the user of the particular client device 16 would be presented with a perhaps reduced frame rate than would otherwise be possible if an increased available bandwidth connection were available and no compression used.

[0041] The battery detect signal, designated S4, reflects the understanding that battery drain is a function of both computational activity and screen activity. A client device dependent on batteries can ill afford a total drain. Transferring an uncompressed bit map image to a display buffer at a client device 16, for example, is much less computationally intensive than decoding and decompressing an MPEG-2 multimedia stream. Such video format (bitmap; uncompressed) would be selected to reduce battery drain. Therefore, video server 12 may incorporate battery management into its decision structure.

[0042] With continued reference to Figure 4, the video format select module 42 is responsive to one or more of the signals S0-S4 for selecting a transmit video format, which is represented by a video format selection signal S5. Applying the foregoing principles, video format select module 42 selects a transmit video format, which is then used by video data formatter module 44 in ensuring compliance of the requested video data with the selected video format. Several examples will now be set forth.

EXAMPLE 1

[0043] Client device 16 includes the client software 46 referred to above, and is identified, for example, as a Windows 98-based machine including 256 MB of SDRAM, and 800 MHz Intel Pentium III central processor, 40 GB hard drive, with a 3D video graphics card having display settings set to 1024 x 768 pixels, 32 bit color depth (true color), connected to the internet via a cable modem having a bandwidth in excess of T1 speeds (*i.e.*, greater than 1.5 Mbps), directly operating from 120 volt line voltage (*i.e.*, no batteries). The requested multimedia content is in an MPEG-2 stored video format. Video format select module 42 makes a determination that the described client device 16 has sufficient computing resources to decode and decompress the requested MPEG-2 format movie in real time. Moreover, any color display characteristics (*i.e.*, color depth) or scaling (*i.e.*, display size) can also be accomplished at the client device 16 in real time. In the described example, the available bandwidth does not impose any limitations, nor is battery usage a factor. While the identification of the client device 16 as a Microsoft Windows-based machine may in some situations counsel in favor of an *.avi video format, server resource usage can be optimized (*i.e.*, no need to make an MPEG-2-to-AVI translation), without any detriment on the perceived experience for the user at the client device end, by the server 12 passing through the retrieved MPEG movie to the requesting client device 16. The transmitted video data is shown in Figure 3 as a message 45 containing video data in an MPEG video format (*i.e.*, the selected transmit video format). No virtual machine 28 is created by server software 26 in this Example 1. The foregoing constitutes one client profile.

EXAMPLE 2

[0044] The client device 16 comprises a so-called Pocket PC having 16 MB of ROM, 32 MB of RAM, and a 240 x 320 TFT LCD screen display with more than 4,000 colors. This client device 16 includes an Ethernet (*i.e.*, a local area network) type connection to network 18, which is coupled to video server

12. This client device 16 runs on batteries. The requested multimedia content is again in an MPEG stored video format. The transmit video format selected by module 42, in this second example, is an AVI video format, in view of the moderate level of computing resources available on such a client device, the display characteristics, and the operating system of the client device (*e.g.*, Microsoft Windows-based operating system). The selection is communicated to video data formatter 44 via signal S5. Video data formatter 44 establishes a virtual machine, for example virtual machine 28_1 in the memory of server 12, for translating from MPEG to AVI. Both MPEG and AVI codecs (coders-decoders) are publicly known and commercially available, as known to those of ordinary skill in the art. Thus, the data stream being retrieved from stored multimedia database 14 is first converted from MPEG into raw video data, and which is then scaled, and whose color depth is further corrected to match that of the client device, and then encoded using an AVI codec. The outgoing stream from video data formatter 44 is sent out via I/O 30 as a stream of messages 45 containing video data in the transmit video format (here, the AVI format). Effectively, much of the processing power required to obtain playback in the client's native format, working from the original compressed MPEG file, is actually borne by the server, not the client. This enhances the client's experience. The foregoing constitutes a second client profile.

EXAMPLE 3

[0045] The client device 16 is a Palm OS connected organizer with a 160 x 160, 16-grey level display, connected by way of a wireless connection (e.g., GSM/PCN, CDMA 2G, 2.5G, etc.). The bandwidth may be 9.6 kbps-14.4 kbps. The client device 16 in this example includes client software 46 for enhanced information gathering and communication upstream to video server 12 through network 18W, 18. Video format select module 42 selects an uncompressed, 16-level grey-scale video format with a scaled size of approximately 160 x 160 pixels for communication to this client 16. The video

format selection is provided via signal S5 to video data formatter 44. Video data formatter 44 initiates another virtual machine 282. As the requested content, which is in an MPEG format, is retrieved from stored multimedia 14, it is, in effect, running (play back) within virtual machine 282. Video data formatter 44, in this example, is configured to convert from color to the 16-level grey scale, as well as to perform a scaling operation to reduce the size so as to be immediately usable for display on the client device 16. In this example, the selected transmit video format is a native video format associated with the described client device 16. All that the client device needs to do is to transfer the incoming, uncompressed (native video format) data from the receive buffer to the display buffer. The foregoing example further illustrates how computing resources are drawn from the device in the chain best adapted to handle (in this example, the server). The foregoing describes a third client profile.

[0046] It should be understood that a whole spectrum of pass through, format conversion, and other variations will be possible depending on all of the factors described above, and yet remain with the spirit and scope of the present invention. In one embodiment, a plurality of client devices 16 of the above-described variety of configurations, are serviced concurrently by video server 12. It should be further understood that video data formatter 44 can be configured to select streaming video feeds that are provided on an alternate input thereto in addition to, or in lieu of, retrieving stored multimedia from stored multimedia database 14. The video server 12 may be adaptive depending on the profile of the client device being serviced in order to provide a better experience to the user. This arrangement overcomes many of the problems in the art, with single purpose or inflexible servers.

[0047] With continued reference to Figure 1, streaming video is provided from a plurality of physically separate locations comprising respective originating video source(s) designated $20_1, \ldots, 20_N$. Each source $20_1, \ldots, 20_N$ include an imaging device 48, a computing device 50 cooperating together to image and stream video data corresponding to a respective view 52.

[0048] Figure 5 shows a single originating source 20_1 in greater detail. Imaging device 48 may be any one of a number of conventional imaging devices, such as still digital cameras, video cameras, and the like. In a constructed embodiment, however, a low-cost web cam is employed, which is a commercially available component, for example, available from D-Link, capable of capturing 30 frames per second, 64,000,000 colors, having a universal serial bus (USB), interface, model no. DSB-C300. Of course, any number of alternative imaging devices, comparable in function, may be substituted therefore.

[0049] Computing device 50_1 may comprise conventional computing apparatus, having a central processing unit (CPU) 54 coupled, as shown in diagrammatic fashion, to a common bus 55, to which communication is enabled with a main memory 56, a display buffer 58, and an input/output (I/O) 60. In a common configuration, image data 62 generated by imaging device 48_1, which corresponds to view 52, is provided to device 50_1 in a compressed format, such as an *.avi video format. Commonly, software provided with imaging devices of the type described (*i.e.*, web cams), allow a user to, among other things, save the incoming video stream 62 to a local hard drive, or to simply allow the image data 62 to be continuously displayed on a locally connected display in a "preview" mode (*i.e.*, the software converts the image data 62 into a bit map and moves the bit map into the display buffer 58). The incoming image data 62, in the *.avi format, for example, is not stored to a local hard drive, but is rather allowed to free run in the "preview" mode. In many computationally, and computing resource weak computing devices 50 it may be difficult to rescale, change from color-to-black and white, or color-to-grey scale, or compress in another format, or do any other conditioning of the image data 62 in real time in addition to the original decompression from AVI format (*i.e.*, a significant portion of the resources have already been consumed converting the incoming image data 62 to a bit map video format for transfer to the display buffer 58).

[0050] Figure 6 shows display buffer 58 in greater detail, having a first portion 66, a portion 68 containing the bit map of the captured view 52, and a remainder portion of the display buffer 70. Accordingly, a module executing from main memory 56 is configured to retrieve the bit map 68 from display buffer 58 and forward the same via I/O 60 as a message 64 containing the bit map data, destined through network 22 to video server 12. It should be understood that Figure 6 is simplified, and that the bit map portion 68 of display buffer 58 need not be linear and contiguous, as shown. Through the foregoing, with a low cost imaging device 48, and a low cost computing device 50, which need not even have a display attached or a keyboard attached thereto, can together cooperate to form a digital image acquisition package. A plurality of such configurations may be deployed at a plurality of distinct physical locations, for example purposes only, for distributed security monitoring. The stream of messages 64 comprising frames in the bit map video format, when routed through video server 12, and when destined for, for example, a relatively low powered client device (*e.g.*, a Palm OS connected organizer, as described above in Example 3), may simply "pass through" the bit map to the client device 16. To that end, the embodiment of Figure 5 can be used in conjunction with the video server to form an end-to-end adaptive video capture and client data server system for improved video communication. A user with a handheld device is then enabled to monitor a plurality of different views at a remote location.

[0051] In this embodiment, a flexible approach to the distribution of multimedia content is provided, wherein resource intensive tasks are allocated to a computing device best suited/best capable of performing such resource intensive tasks. If the video server is the node in the system having the greatest computing power, most of the conditioning work on the video data is performed there. If the client has extensive capabilities, then the video server just serves up the content, which is usually compressed (MPEG, AVI), and allows the client to decode and decompress it. Likewise, if the originating source (*i.e.*, image or video capture source) is resource limited, a bitmap only

is extracted and sent upstream to the video server. The system described herein provides a comprehensive approach applicable across many of the common client devices today, and is particularly suited for wireless, and/or handheld computing devices.

CLAIMS

1. A client-server system comprising:

a client configured to display video coupled to a network;

a server coupled to said network configured to determine computing resource characteristics of said client, said server being further configured to select a transmit video format from a plurality of video formats based on said determined characteristics and to communicate video data to said client in said transmit video format.

2. The system of claim 1 wherein said video data is stored on said server in a stored video format.

3. The system of claim 2 wherein said stored video format is the same as said transmit video format.

4. The system of claim 3 wherein said stored video format is a Motion Picture Experts Group (MPEG) compliant video format.

5. The system of claim 4 wherein said stored video format is an MPEG-2 compliant format.

6. The system of claim 2 wherein said stored video format is different than said transmit video format.

7. The system of claim 6 wherein said stored video format is a Motion Picture Experts Group (MPEG) compliant video format and said transmit format is a native video format associated with said client.

8. The system of claim 1 wherein said computing resource characteristics is selected from the group comprising central processing unit (CPU) computing ability, an amount of main memory, an amount of available memory, and an availability of a specialized graphics processor.

9. The system of claim 1 wherein said server is configured to determine display characteristics of said client, said server being configured to select said transmit video format further as a function of said display characteristic.

10. The system of claim 9 wherein said display characteristic is selected from the group comprising an availability of color display and color depth associated therewith, an availability of grey-scale display and a number of levels of gray associated therewith, an availability of black/white display, and a horizontal and vertical display size.

11. The system of claim 1 wherein said server is configured to determine a bandwidth through said network between said server and said client, said server being configured to select said transmit video format further as a function of said determined bandwidth.

12. The system of claim 1 wherein said video data is fed to said server from an originating source.

13. The system of claim 12 wherein said originating source comprises:

a computing device having a main memory and a video buffer; and

an imaging device coupled to said computing device;

said computing device being configured to receive image data from said imaging device of a first view and store a bitmap of said first view in a portion of said video buffer, said computing device being further configured to communicate said portion of said video buffer to said server.

14. The system of claim 12 wherein said video data is fed to said server from a plurality of originating sources.

15. The system of claim 12 wherein said server is configured to transmit said video data to a plurality of clients.

16. The system of claim 12 wherein said plurality of sources are disposed in at least two different physical locations.

17. The system of claim 1 wherein said server is configured to determine an identity of said client, said server being configured to select said transmit video format further as a function of said determined identity.

18. The system of claim 1 wherein said server is configured to transmit video data to a plurality of clients wherein a first client differs from a second client in at least one of computing resource characteristic, display characteristic, bandwidth through said network to said server, or identity.

19. The system of claim 18 wherein said server is further configured to transmit said video data to said plurality of client concurrently.

20. The system of claim 1 wherein said server is configured to determine (i) display characteristics of said client, (ii) a bandwidth through said network between said server and said client, and (iii) an identity of said client, said server being configured to select said transmit video format further as a function of said determined display characteristic, said determined bandwidth, and said determined identity.

21. The system of claim 1 wherein said server is configured to determine a battery characteristic associated with said client, said server being further configured to select said transmit format further as a function of said determined battery characteristic.

22. A client-server system comprising:

a client configured to display video coupled to a network;

a server coupled to said network configured to determine computing resource characteristics of said client available for processing compressed video, said server being configured to select a transmit video format from a plurality of video formats based on said determined characteristics and to communicate video data to said client in said transmit video format.

23. The system of claim 22 wherein said server is configured to determine at least one of (i) a display characteristic of said client, (ii) a bandwidth through said network between said server and said client, (iii) an identity of said client, and (iv) a battery characteristic associated with said client, said server being further configured to select said transmit format further as a function of at least one of said determined display characteristic, said determined bandwidth, said determined identity or said determined battery characteristic.

24. The system of claim 23 wherein said video data is fed to said server from an originating source, said originating source comprising:

a computing device having a main memory and a video buffer; and

an imaging device coupled to said computing device;

said computing device being configured to receive image data from said imaging device of a first view and store a bitmap of said first view in a portion of said video buffer, said computing device being further configured to communicate said portion of said video buffer to said server in said bitmap format.

25. A method of communicating video data over a network to a client comprising:

determining a computing resource characteristic of said client; and

selecting a transmit video format based on the determined computing resource characteristic of said client.

26. The method of claim 25 wherein the transmit video format is selected based on at least one of a display characteristic of said client, an identity of said client or an available bandwidth to said client.

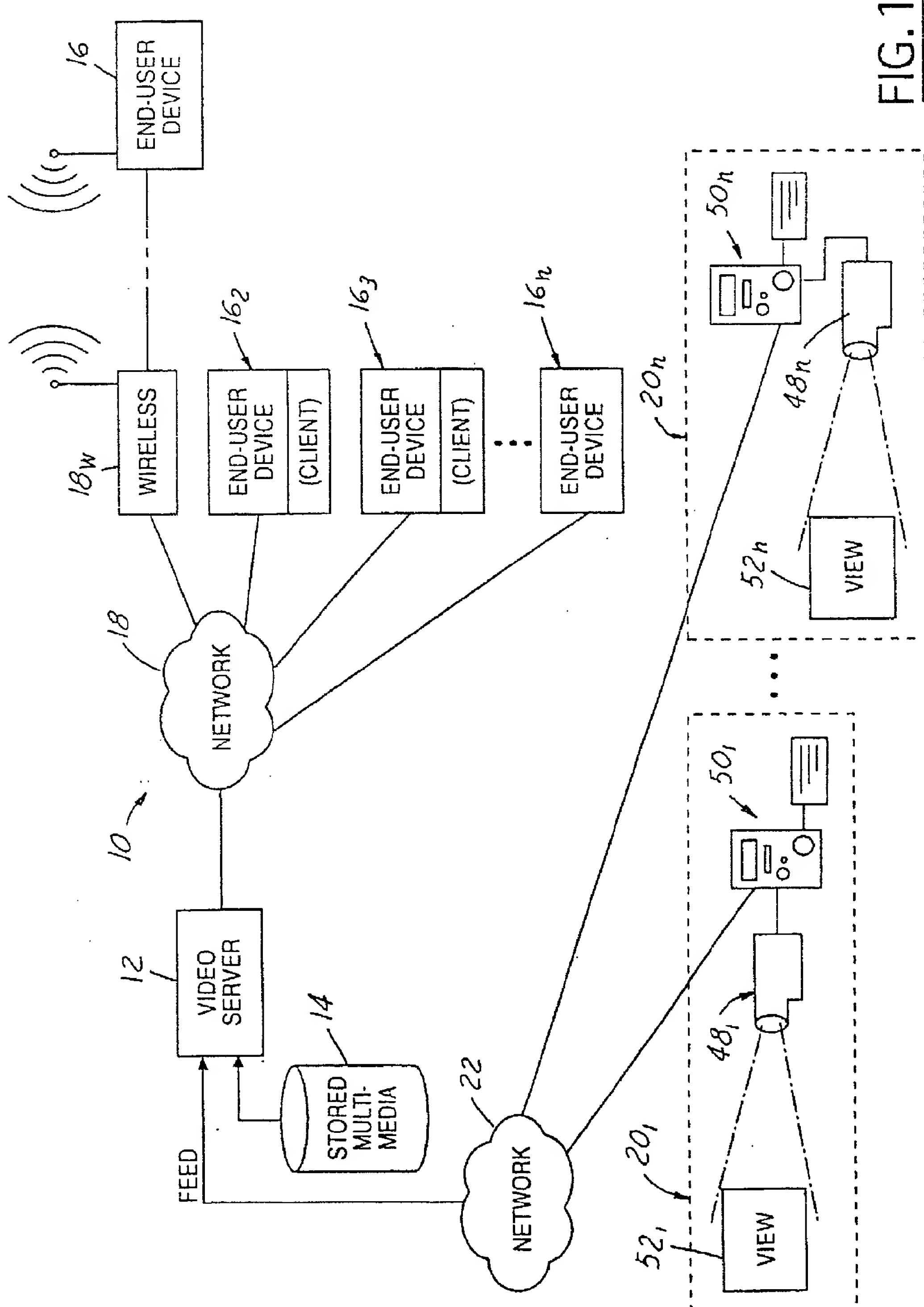
END-USER DEVICE
16
WIRELESS
18W
END-USER DEVICE
(CLIENT)
162
163
16n
NETWORK
18
10
VIDEO SERVER
12
STORED MULTI-MEDIA
14
FEED
NETWORK
22
201
20n
501
50n
481
48n
VIEW
521
52n

FIG. 1

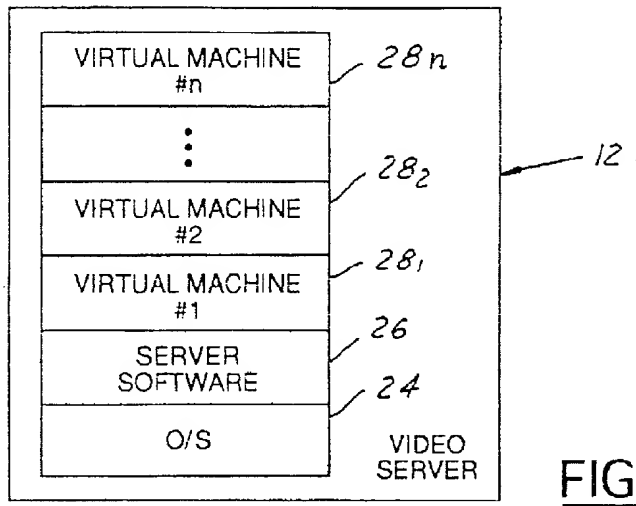
VIRTUAL MACHINE #n
28n
VIRTUAL MACHINE #2
28_2
VIRTUAL MACHINE #1
28_1
SERVER SOFTWARE
26
O/S
24
VIDEO SERVER
12

FIG. 2

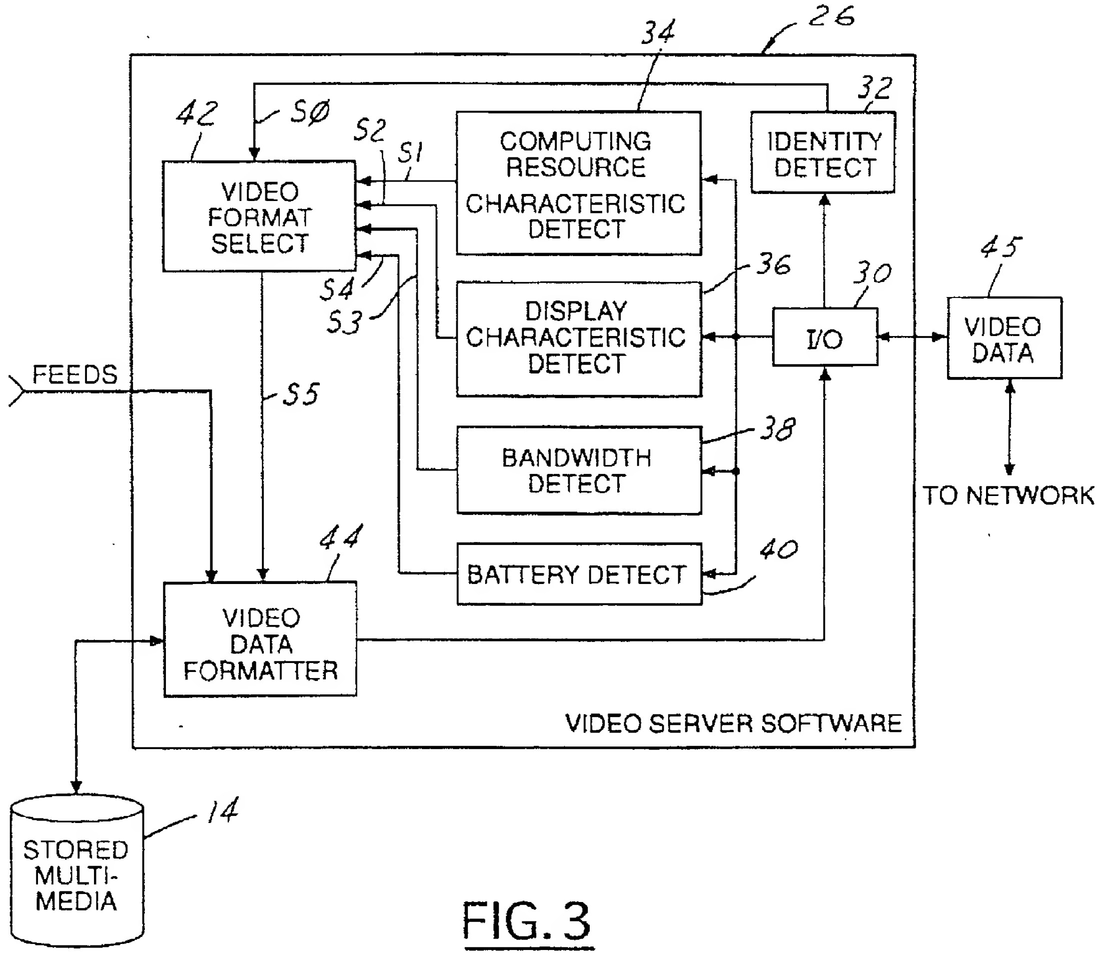
26
34
32
42
SØ
S2
S1
VIDEO FORMAT SELECT
COMPUTING RESOURCE CHARACTERISTIC DETECT
IDENTITY DETECT
S4
S3
36
30
45
DISPLAY CHARACTERISTIC DETECT
I/O
VIDEO DATA
FEEDS
S5
BANDWIDTH DETECT
38
TO NETWORK
44
40
BATTERY DETECT
VIDEO DATA FORMATTER
VIDEO SERVER SOFTWARE
STORED MULTI-MEDIA
14

FIG. 3

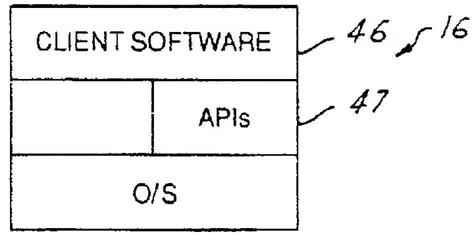

FIG. 4

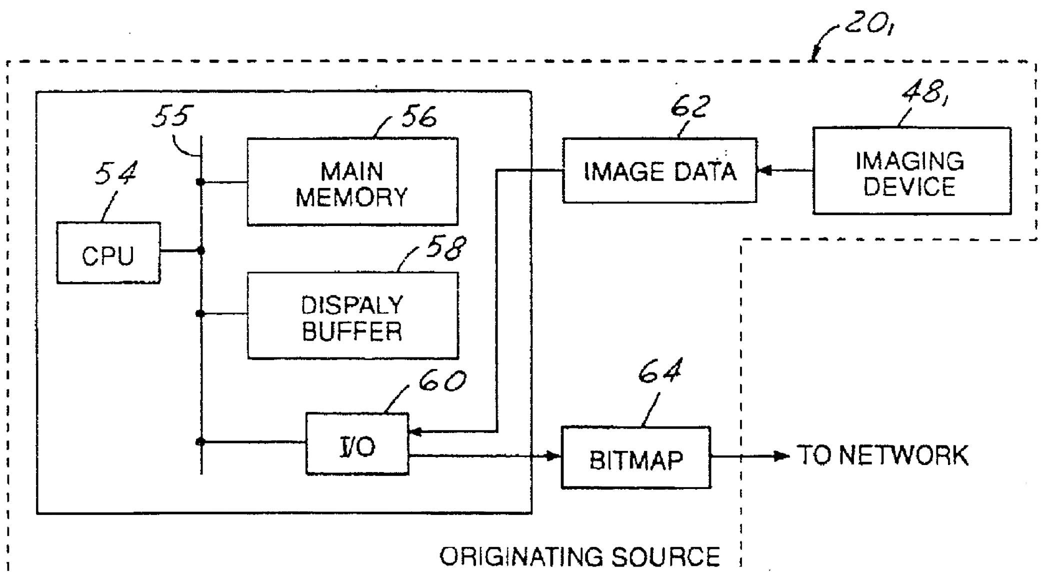

FIG. 5

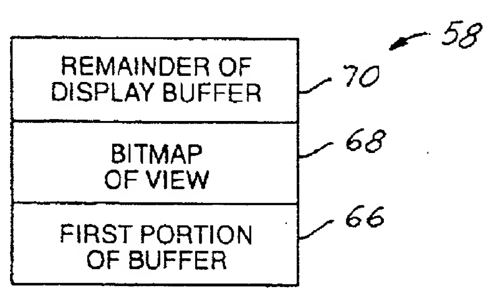

FIG. 6